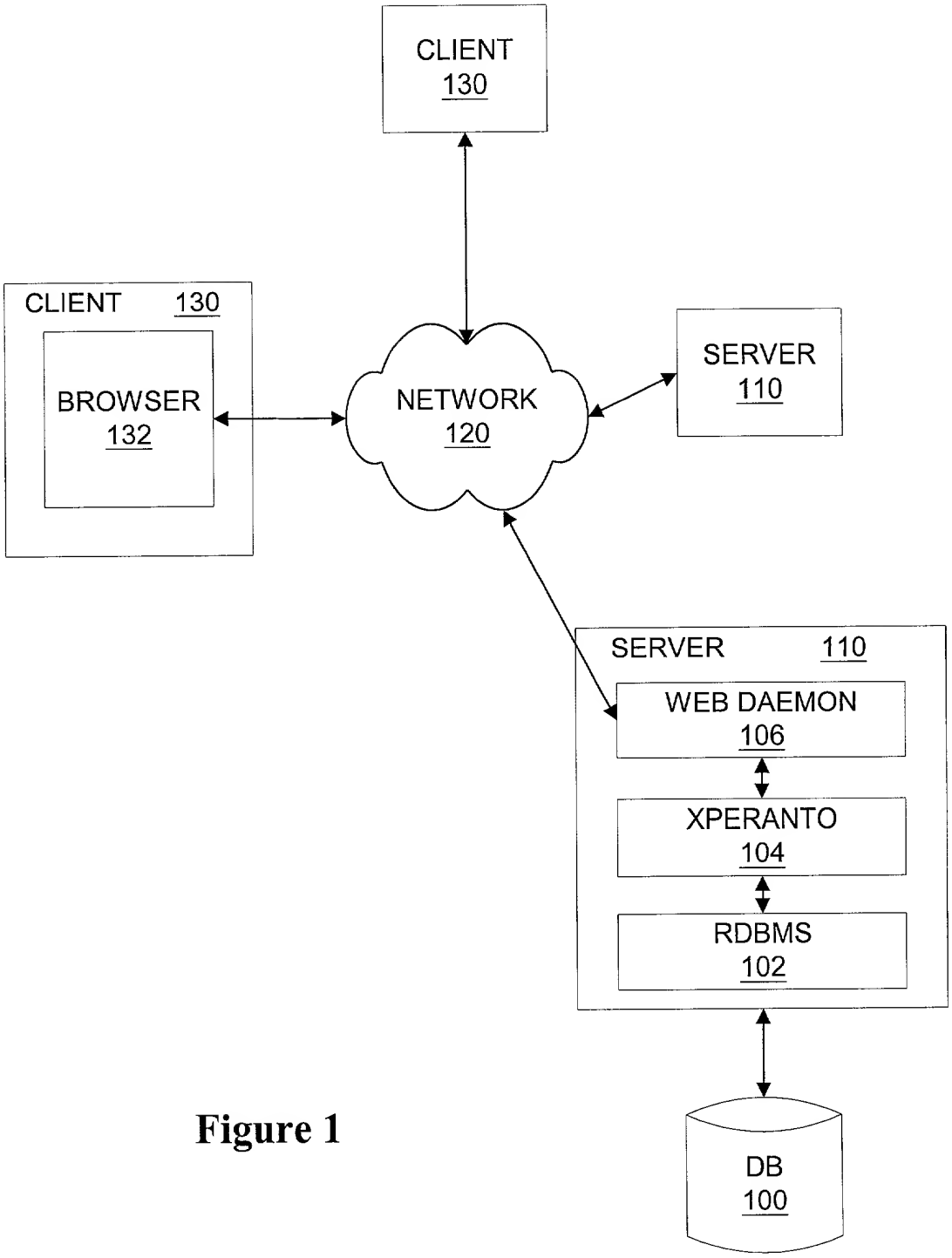
CLIENT
130
CLIENT 130
BROWSER
132
NETWORK
120
SERVER
110
SERVER 110
WEB DAEMON
106
XPERANTO
104
RDBMS
102
DB
100

Figure 1

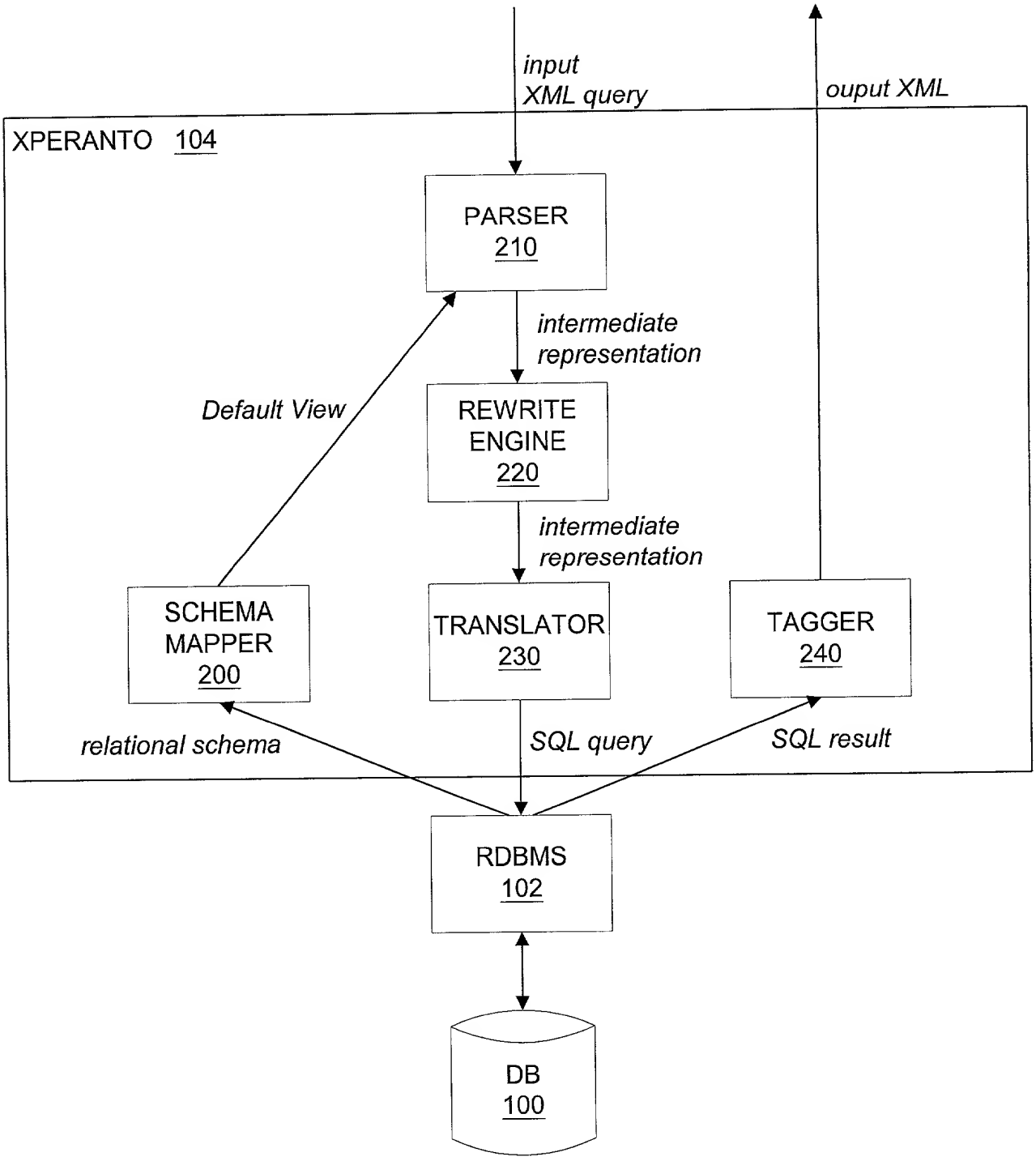
input
XML query
ouput XML
XPERANTO 104
PARSER
210
intermediate
representation
Default View
REWRITE
ENGINE
220
intermediate
representation
SCHEMA
MAPPER
200
TRANSLATOR
230
TAGGER
240
relational schema
SQL query
SQL result
RDBMS
102
DB
100

Figure 2

DEPT TABLE 300

DNO	NAME	SIZE
1	Finance	100
2	Marketing	200
3	Sales	50
...		

Figure 3

EMP TABLE 310

DNO	ENO	NAME	AGE
1	1	J. Skolem	50
2	2	L. Hubbard	33
...			

Default XML View of the DEPT Table 320

```
<DEPT>
  <ROW>
    <DNO>1</DNO><NAME>Finance</NAME><SIZE>100</SIZE>
  </ROW>
  <ROW>
    <DNO>2</DNO><NAME>Marketing</NAME><SIZE>200</SIZE>
  </ROW>
  <ROW>
    <DNO>3</DNO><NAME>Sales</NAME><SIZE>50</SIZE>
  </ROW>
  ...
</DEPT>
```

Default XML View of the EMP Table 330

```
<EMP>
  <ROW>
    <DNO>1</DNO><ENO>1</ENO><NAME>J. Skolem</NAME><AGE>50</AGE>
  </ROW>
  <ROW>
    <DNO>2</DNO><ENO>2</ENO><NAME>L. Hubbard</NAME><AGE>33</AGE>
  </ROW>
  ...
</EMP>
```

Query Over the Default View

```
WHERE
  <DEPT>
    <ROW>
      <DNO>$DNO</DNO><NAME>$DNAME</NAME>
      <SIZE>$DSIZE</SIZE>
    </ROW>
  </DEPT>IN DefaultView,
  $DSIZE>75
CONSTRUCT
  <BIGDEPT>
    <NAME>$DNAME</NAME>
    <SIZE>$DSIZE</SIZE>
    {WHERE
      <EMP>
        <ROW><DNO>$DNO</DNO><NAME>$ENAME</NAME></ROW>
      </EMP>IN DefaultView
    CONSTRUCT
      <EMP><NAME>$ENAME</NAME></EMP>}
  </BIGDEPT>
```

XML Produced by the Query

```
<BIGDEPT>
  <NAME>Finance</NAME>
  <SIZE>100</SIZE>
  <EMP><NAME>J.Skolem</NAME></EMP>
  ...
</BIGDEPT>
<BIGDEPT>
  <NAME>Marketing</NAME>
  <SIZE>200</SIZE>
  <EMP><NAME>L. Hubbard</NAME></EMP>
  ...
</BIGDEPT>
...
```

Figure 4

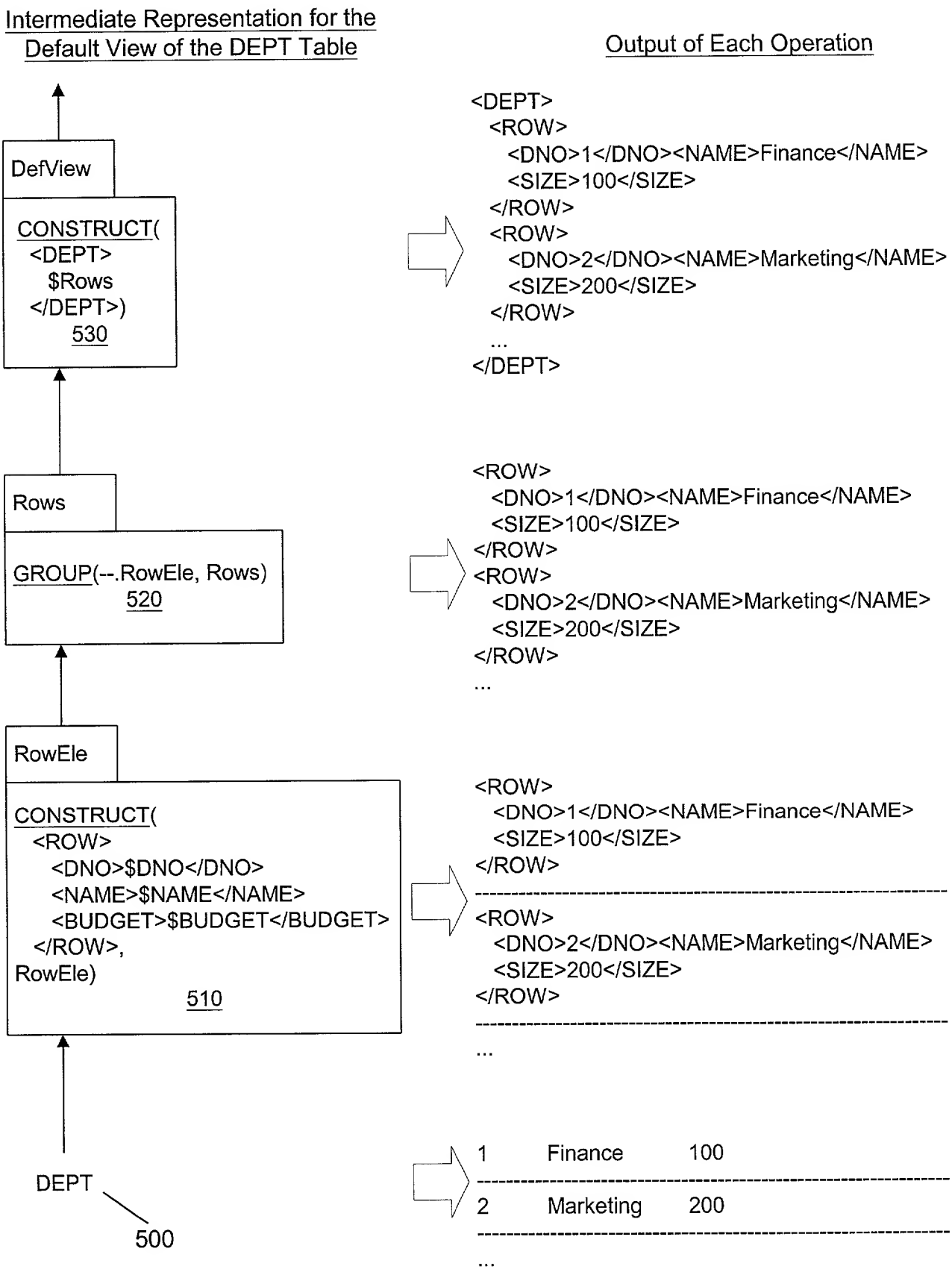
Intermediate Representation for the Default View of the DEPT Table
Output of Each Operation
DefView
CONSTRUCT(
<DEPT>
$Rows
</DEPT>)
530
Rows
GROUP(--.RowEle, Rows)
520
RowEle
CONSTRUCT(
<ROW>
<DNO>$DNO</DNO>
<NAME>$NAME</NAME>
<BUDGET>$BUDGET</BUDGET>
</ROW>,
RowEle)
510
DEPT
500
<DEPT>
<ROW>
<DNO>1</DNO><NAME>Finance</NAME>
<SIZE>100</SIZE>
</ROW>
<ROW>
<DNO>2</DNO><NAME>Marketing</NAME>
<SIZE>200</SIZE>
</ROW>
...
</DEPT>
<ROW>
<DNO>1</DNO><NAME>Finance</NAME>
<SIZE>100</SIZE>
</ROW>
<ROW>
<DNO>2</DNO><NAME>Marketing</NAME>
<SIZE>200</SIZE>
</ROW>
...
<ROW>
<DNO>1</DNO><NAME>Finance</NAME>
<SIZE>100</SIZE>
</ROW>
<ROW>
<DNO>2</DNO><NAME>Marketing</NAME>
<SIZE>200</SIZE>
</ROW>
...
1 Finance 100
2 Marketing 200
...

Figure 5

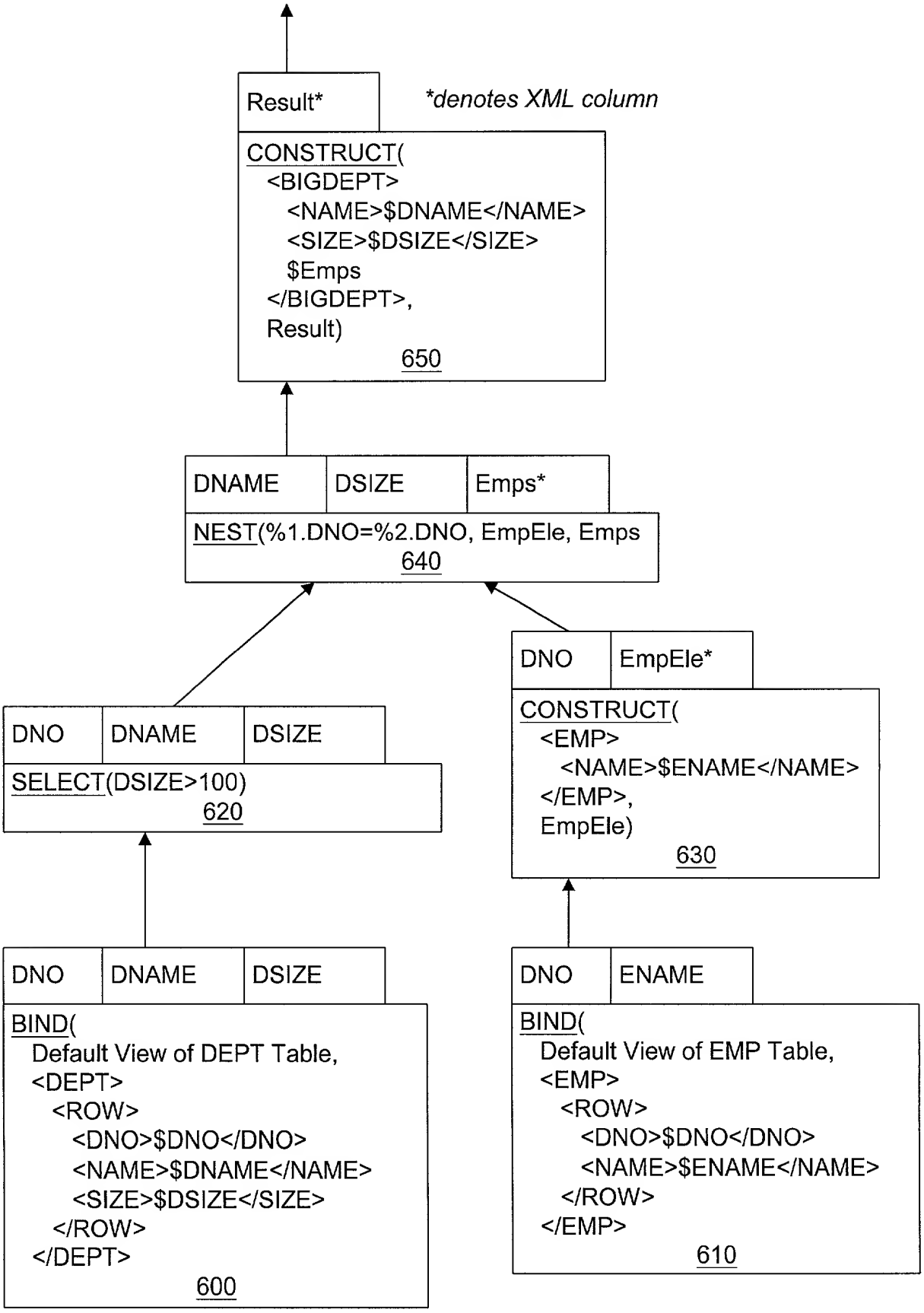
Result*
*denotes XML column
CONSTRUCT(
<BIGDEPT>
<NAME>$DNAME</NAME>
<SIZE>$DSIZE</SIZE>
$Emps
</BIGDEPT>,
Result)
650
DNAME
DSIZE
Emps*
NEST(%1.DNO=%2.DNO, EmpEle, Emps
640
DNO
DNAME
DSIZE
SELECT(DSIZE>100)
620
DNO
EmpEle*
CONSTRUCT(
<EMP>
<NAME>$ENAME</NAME>
</EMP>,
EmpEle)
630
DNO
DNAME
DSIZE
BIND(
Default View of DEPT Table,
<DEPT>
<ROW>
<DNO>$DNO</DNO>
<NAME>$DNAME</NAME>
<SIZE>$DSIZE</SIZE>
</ROW>
</DEPT>
600
DNO
ENAME
BIND(
Default View of EMP Table,
<EMP>
<ROW>
<DNO>$DNO</DNO>
<NAME>$ENAME</NAME>
</ROW>
</EMP>
610

Figure 6

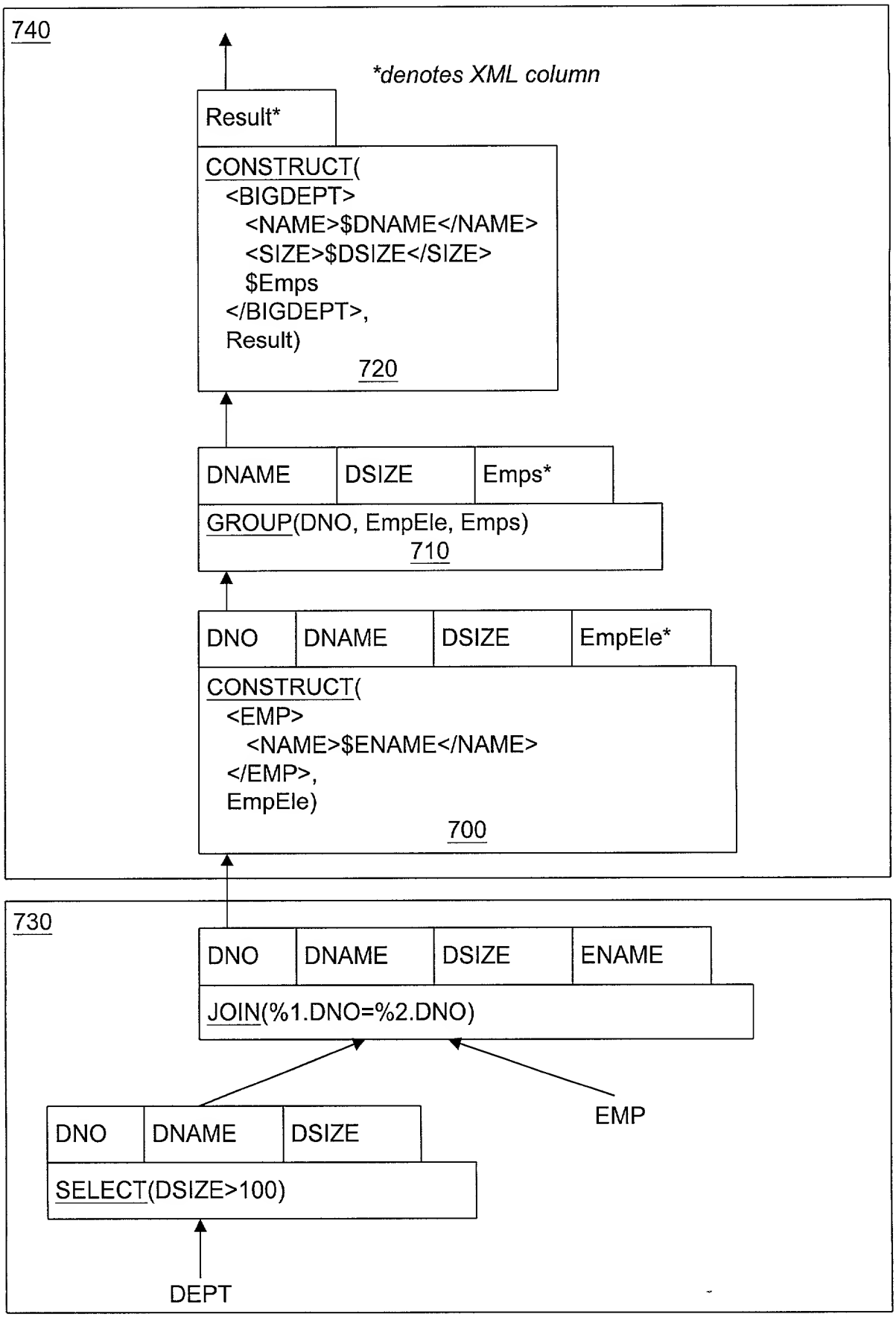
740
*denotes XML column
Result*
CONSTRUCT(
<BIGDEPT>
<NAME>$DNAME</NAME>
<SIZE>$DSIZE</SIZE>
$Emps
</BIGDEPT>,
Result)
720
DNAME
DSIZE
Emps*
GROUP(DNO, EmpEle, Emps)
710
DNO
DNAME
DSIZE
EmpEle*
CONSTRUCT(
<EMP>
<NAME>$ENAME</NAME>
</EMP>,
EmpEle)
700
730
DNO
DNAME
DSIZE
ENAME
JOIN(%1.DNO=%2.DNO)
EMP
DNO
DNAME
DSIZE
SELECT(DSIZE>100)
DEPT

Figure 7

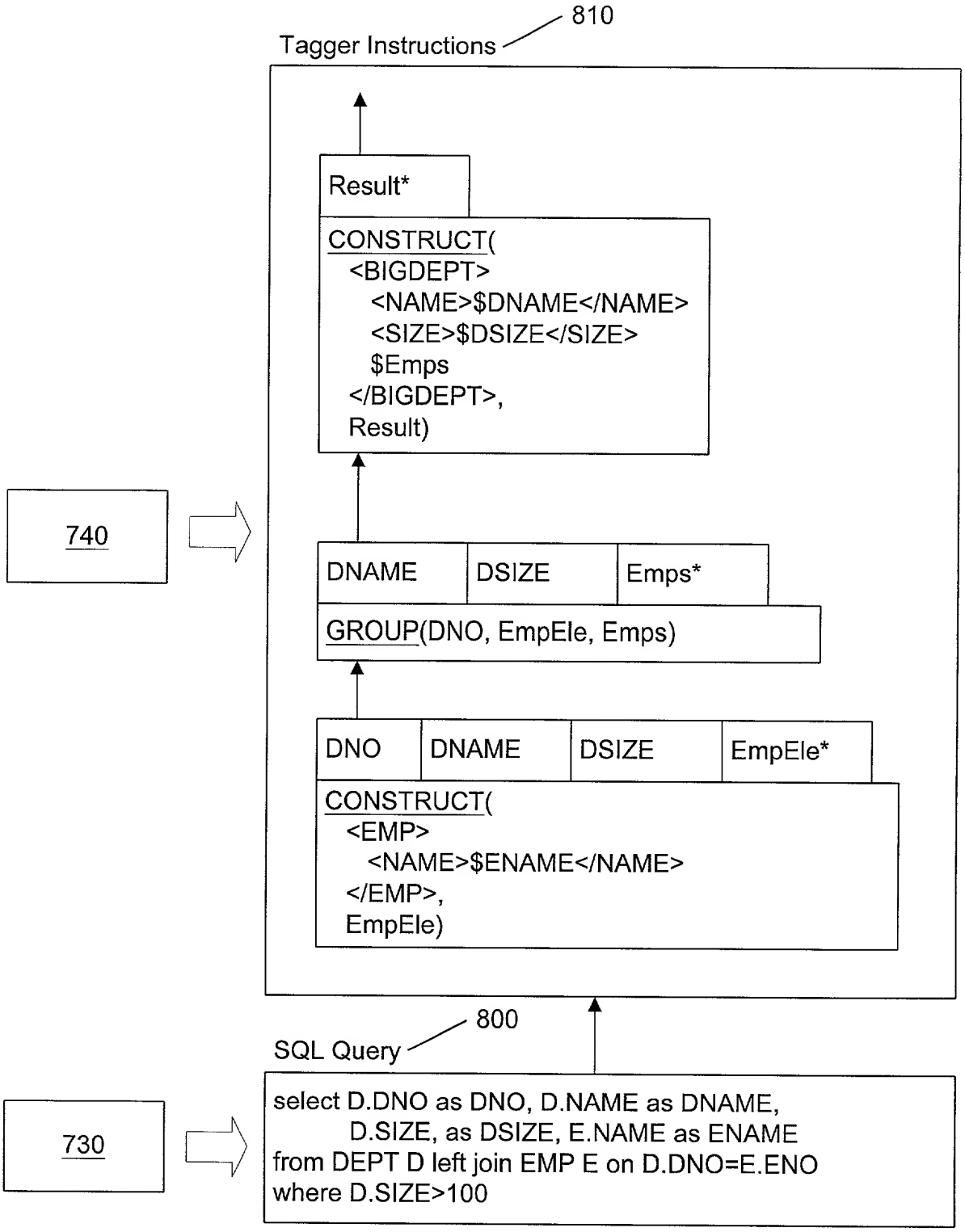
810
Tagger Instructions
Result*
CONSTRUCT(
<BIGDEPT>
<NAME>$DNAME</NAME>
<SIZE>$DSIZE</SIZE>
$Emps
</BIGDEPT>,
Result)
740
DNAME
DSIZE
Emps*
GROUP(DNO, EmpEle, Emps)
DNO
DNAME
DSIZE
EmpEle*
CONSTRUCT(
<EMP>
<NAME>$ENAME</NAME>
</EMP>,
EmpEle)
800
SQL Query
730
select D.DNO as DNO, D.NAME as DNAME,
D.SIZE, as DSIZE, E.NAME as ENAME
from DEPT D left join EMP E on D.DNO=E.ENO
where D.SIZE>100

Figure 8

Employee

WORKDEPT	ENO	LASTNAME
1	1	Haas
2	2	Thompson
3	3	Kwan
1	4	Lucchessi
1	5	O'Connell
3	6	Quintana
3	7	Nicholls

Figure 9a

Default XML View of the Employee Table (DefaultEmployee)

```
<EMP>
 <ROW>
   <WORKDEPT>1</WORKDEPT><ENO>1</ENO><LASTNAME>Haas</LASTNAME>
 </ROW>
 <ROW>
   <WORKDEPT>2</WORKDEPT><ENO>2</ENO><LASTNAME>Thompson</LASTNAME>
 </ROW>
 <ROW>
   <WORKDEPT>3</WORKDEPT><ENO>3</ENO><LASTNAME>Kwan</LASTNAME>
 </ROW>
 <ROW>
   <WORKDEPT>1</WORKDEPT><ENO>4</ENO><LASTNAME>Lucchessi</LASTNAME>
 </ROW>
 <ROW>
   <WORKDEPT>1</WORKDEPT><ENO>5</ENO><LASTNAME>O'Connell</LASTNAME>
 </ROW>
 <ROW>
   <WORKDEPT>3</WORKDEPT><ENO>6</ENO><LASTNAME>Quintana</LASTNAME>
 </ROW>
 <ROW>
   <WORKDEPT>3</WORKDEPT><ENO>7</ENO><LASTNAME>Nicholls</LASTNAME>
 </ROW>
</EMP>
```

Figure 9b

Department

DEPTNO	DEPTNAME	SIZE
1	Spiffy Computer	3
2	Planning	1
3	Information Center	3
4	Development Center	0

Figure 10a

Default XML View of the Department Table (DefaultDepartment)

```
<DEPT>
        <ROW>
                <DEPTNO>1</DEPTNO><DEPTNAME>Spiffy Computer </DEPTNAME><SIZE>3</SIZE>
        </ROW>
        <ROW>
                <DEPTNO>2</DEPTNO><DEPTNAME>Planning</DEPTNAME><SIZE>1</SIZE>
        </ROW>
        <ROW>
                <DEPTNO>3</DEPTNO><DEPTNAME>Information
Center</DEPTNAME><SIZE>3</SIZE>
        </ROW>
        <ROW>
                <DEPTNO>4</DEPTNO><DEPTNAME>Development
Center</DEPTNAME><SIZE>0</SIZE>
        </ROW>
</DEPT>
```

Figure 10b

Query Over Default Views

```
WHERE
    <DEPT>
        <ROW>
            <DEPTNO>$DEPTNO</DEPTNO> <DEPTNAME>$DEPTNAME</DEPTNAME>
            <SIZE>$SIZE</SIZE>
        </ROW>
    </DEPT> IN DefaultDepartment
CONSTRUCT
  <dept name=$DEPTNAME>
    <emplist>
     {WHERE
          <ROW>
             <WORKDEPT>$DEPTNO</WORKDEPT><ENO>$ENO</ENO>
             <LASTNAME>$ENAME</LASTNAME>
          </ROW>
       </EMP> IN DefaultEmployee
      CONSTRUCT
        <employee> <name> $ENAME </name> </employee>
     }
    </emplist>
  </dept>
```

Figure 11a

Result of Query

```
<dept name="SPIFFY COMPUTER">
        <emplist>
                <employee><name>HAAS</name></employee>
                <employee><name>LUCCHESSI</name></employee>
                <employee><name>O'CONNELL</name> </employee>
  </emplist>
</dept>
<dept name="PLANNING">
        <emplist>
                <employee> <name>THOMPSON</name></employee>
  </emplist>
 </dept>
 <dept name="INFORMATION CENTER">
        <emplist>
                 <employee><name>KWAN</name></employee>
                 <employee><name>QUINTANA</name></employee>
                 <employee><name>NICHOLLS</name></employee>
        </emplist>
 </dept>
<dept name="DEVELOPMENT CENTER">
  <emplist >
        </emplist>
 </dept>
```

Figure 11b

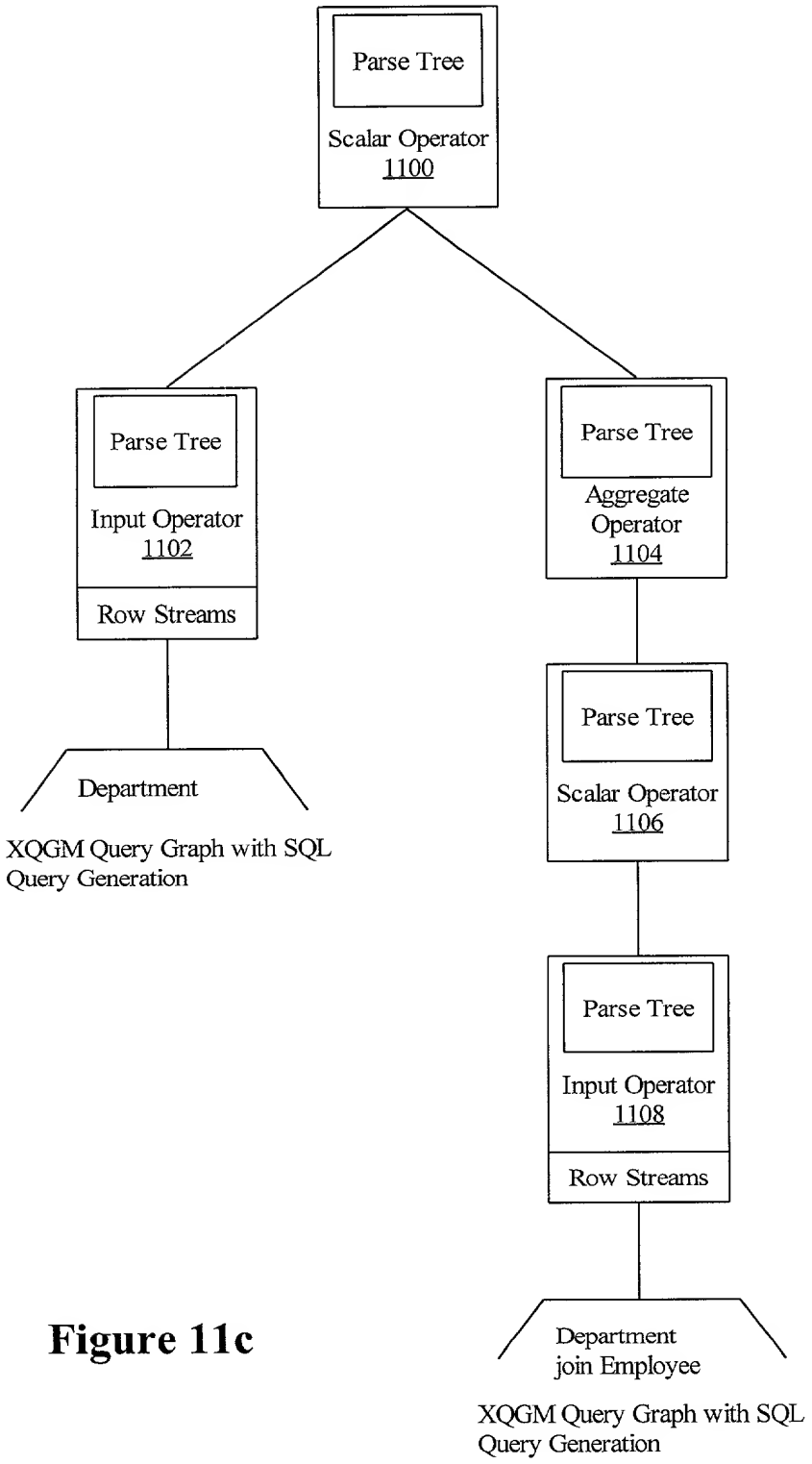
Parse Tree
Scalar Operator
1100
Parse Tree
Input Operator
1102
Row Streams
Department
XQGM Query Graph with SQL
Query Generation
Parse Tree
Aggregate
Operator
1104
Parse Tree
Scalar Operator
1106
Parse Tree
Input Operator
1108
Row Streams
Department
join Employee
XQGM Query Graph with SQL
Query Generation

Figure 11c

Sorted Outer Union SQL Query

```
select q4.c2, q4.c3, q4.c4
from   (select q3.DEPTNO, q3.DEPTNAME, cast (null as VARCHAR(15)), 0
       from DEPARTMENT AS q3

       union all
       select q2.DEPTNO, cast (null as VARCHAR(29)), q2.LASTNAME, 1
       from   (select q1.LASTNAME, q5.DEPTNO
              from   EMPLOYEE AS q1,
                     DEPARTMENT AS q5
              where  (q5.DEPTNO = q1.WORKDEPT))
       AS q2(LASTNAME, DEPTNO))
AS Q4(DEPTNO, DEPTNAME, LASTNAME, INDICATOR)
order by DEPTNO, INDICATOR
```

2

1

Figure 12a

Result of 1

DEPTNO	'NULL'	LASTNAME	'INDICATOR'
1	null	Haas	1
2	null	Thompson	1
3	null	Kwan	1
1	null	Lucchessi	1
1	null	O'Connell	1
3	null	Quintana	1
3	null	Nicholls	1

Figure 12b

Result of 2

DEPTNO	DEPTNAME	'NULL'	'INDICATOR'
1	Spiffy Computer	null	0
2	Planning	null	0
3	Information Center	null	0
4	Development Center	null	0

Figure 12c

Result of Union

DEPTNO	DEPTNAME	LASTNAME	INDICATOR
1	Spiffy Computer	null	0
2	Planning	null	0
3	Information Center	null	0
4	Development Center	null	0
1	null	Haas	1
2	null	Thompson	1
3	null	Kwan	1
1	null	Lucchessi	1
1	null	O'Connell	1
3	null	Quintana	1
3	null	Nicholls	1

Figure 12d

Result of Order

DEPTNO	DEPTNAME	LASTNAME	INDICATOR
1	Spiffy Computer	null	0
1	null	Haas	1
1	null	Lucchessi	1
1	null	O'Connell	1
2	Planning	null	0
2	null	Thompson	1
3	Information Center	null	0
3	null	Kwan	1
3	null	Quintana	1
3	null	Nicholls	1
4	Development Center	null	0

Figure 12e

Result of Sorted Outer Union Query

DEPTNAME	LASTNAME	INDICATOR
Spiffy Computer	null	0
null	Haas	1
null	Lucchessi	1
null	O'Connell	1
Planning	null	0
null	Thompson	1
Information Center	null	0
null	Kwan	1
null	Quintana	1
null	Nicholls	1
Development Center	null	0

Figure 12f

Node Strip SQL Query

```
select q1.DEPTNO, q1.DEPTNAME
from   DEPARTMENT AS q1                          } 1
order by q1.DEPTNO

select q2.LASTNAME, q2.DEPTNO
from   (select q3.LASTNAME, q4.DEPTNO
       from   EMPLOYEE AS q3,
              DEPARTMENT AS q4                   } 2
       where  (q4.DEPTNO = q3.WORKDEPT))
AS q2(LASTNAME, DEPTNO)
order by q2.DEPTNO
```

Figure 13a

Result of Node Strip (1)

DEPTNO	DEPTNAME
1	Spiffy Computer
2	Planning
3	Information Center
4	Development Center

Figure 13b

Result of Node Strip (2)

LASTNAME	DEPTNO
Haas	1
Lucchessi	1
O'Connell	1
Thompson	2
Kwan	3
Quintana	3
Nicholls	3

Figure 13c

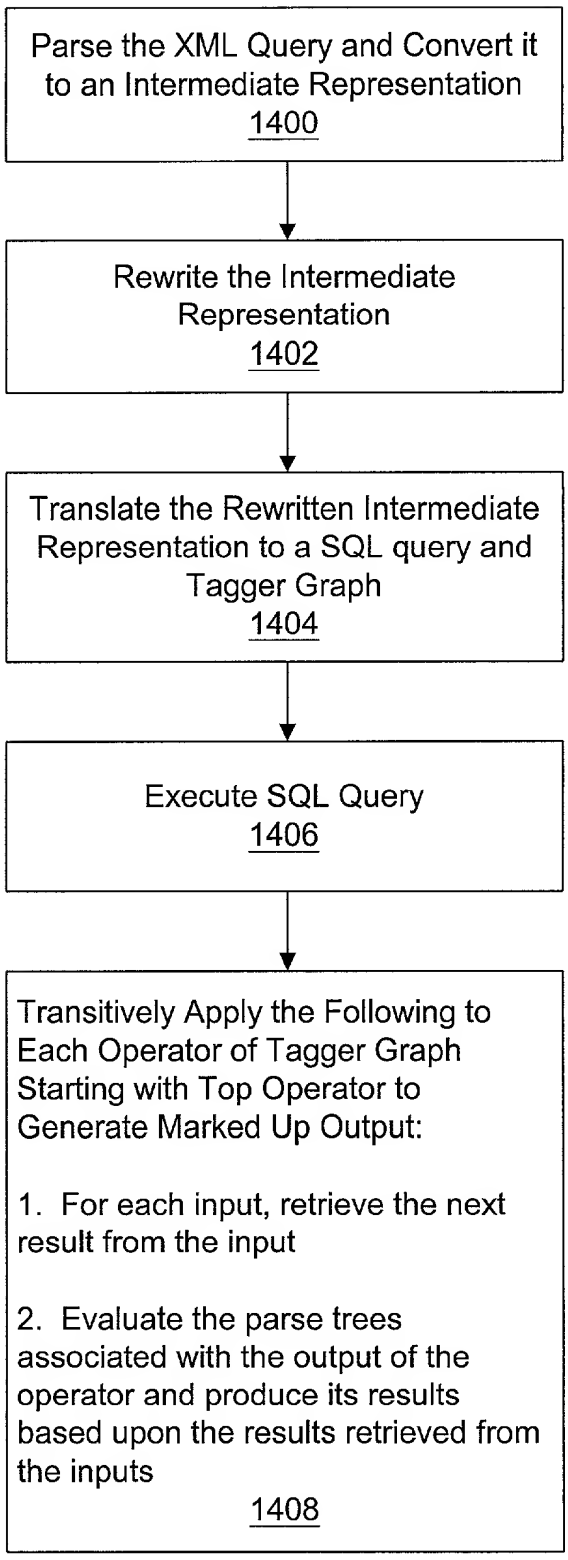
Parse the XML Query and Convert it to an Intermediate Representation
1400
Rewrite the Intermediate Representation
1402
Translate the Rewritten Intermediate Representation to a SQL query and Tagger Graph
1404
Execute SQL Query
1406
Transitively Apply the Following to Each Operator of Tagger Graph Starting with Top Operator to Generate Marked Up Output:
1. For each input, retrieve the next result from the input
2. Evaluate the parse trees associated with the output of the operator and produce its results based upon the results retrieved from the inputs
1408

Figure 14